# Concurrency Theory

Winter 2025/26

Lecture 5: Game Characterisation and Variants of Strong Bisimulation

Thomas Noll, Peter Thiemann
Programming Languages Group
University of Freiburg

https://proglang.github.io/teaching/25ws/ct.html

Thomas Noll, Peter Thiemann

Winter 2025/26

# Strong Bisimulation

**Definition (Strong bisimulation)** (Park 1981, Milner 1989)

A binary relation $\rho \subseteq Prc \times Prc$ is a strong bisimulation whenever for every $(P, Q) \in \rho$ and $\alpha \in Act$:

(1) if $P \xrightarrow{\alpha} P'$, then there exists $Q' \in Prc$ such that $Q \xrightarrow{\alpha} Q'$ and $P' \,\rho\, Q'$, and

(2) if $Q \xrightarrow{\alpha} Q'$, then there exists $P' \in Prc$ such that $P \xrightarrow{\alpha} P'$ and $P' \,\rho\, Q'$.

**Note:** strong bisimulations are not necessarily equivalences (e.g., $\rho = \emptyset$).

**Definition (Strong bisimilarity)**

Processes $P, Q \in Prc$ are strongly bisimilar, denoted $P \sim Q$, iff there is a strong bisimulation $\rho$ with $P \,\rho\, Q$.

$$\sim \; = \bigcup \{\rho \subseteq Prc \times Prc \mid \rho \text{ is a strong bisimulation}\}.$$

Relation $\sim$ is called strong bisimilarity.

# Properties of Strong Bisimilarity

## Lemma (Properties of $\sim$)

(1) $\sim$ *is an equivalence relation (i.e., reflexive, symmetric, and transitive).*

(2) $\sim$ *is the coarsest strong bisimulation.*

## Proof.

(1) $\sim$ is an equivalence relation:

- Reflexivity:

$$\mathrm{id}_{Prc} := \{(P, P) \mid P \in Prc\}$$

  is obviously a strong bisimulation.
  Since $\mathrm{id}_{Prc} \subseteq \sim$ by Definition 4.2, $\sim$ is reflexive.
- Symmetry: (**Caveat:** not every strong bisimulation is symmetric; cf. Example 4.4.)
  But if $\rho$ is a strong bisimulation, then so is its inverse

$$\rho^{-1} := \{(Q, P) \mid P\rho Q\}$$

# Properties of Strong Bisimilarity

## Lemma (Properties of $\sim$)

(1) $\sim$ *is an equivalence relation (i.e., reflexive, symmetric, and transitive).*

(2) $\sim$ *is the coarsest strong bisimulation.*

## Proof.

(1) $\sim$ is an equivalence relation:

- Transitivity: (**Caveat:** not every strong bisimulation is transitive.) But if $\rho$ and $\sigma$ are strong bisimulations, then so is their composition

$$\rho \circ \sigma := \{(P, R) \mid \exists Q : P\rho Q, Q\sigma R\}.$$

  Proof: $P\ (\rho \circ \sigma)\ R$ and $P \xrightarrow{\alpha} P'$
  $\Rightarrow \exists Q : P\rho Q, Q\sigma R$ and $P \xrightarrow{\alpha} P'$ (def. $\circ$)
  $\Rightarrow \exists Q, Q' : Q\sigma R, Q \xrightarrow{\alpha} Q'$ and $P' \rho Q'$ ($\rho$ strong bisimulation)
  $\Rightarrow \exists Q', R' : P' \rho Q', R \xrightarrow{\alpha} R'$ and $Q' \sigma R'$ ($\sigma$ strong bisimulation)
  $\Rightarrow \exists R' : R \xrightarrow{\alpha} R'$ and $P'\ (\rho \circ \sigma)\ R'$ (def. $\circ$)
  (analogously for assumption $R \xrightarrow{\alpha} R'$)

# Properties of Strong Bisimilarity

**Lemma (Properties of $\sim$)**

(1) *$\sim$ is an equivalence relation (i.e., reflexive, symmetric, and transitive).*

(2) *$\sim$ is the coarsest strong bisimulation.*

**Proof.**

(2) $\sim$ is the coarsest strong bisimulation:
According to Definition 4.2, it suffices to show that strong bisimulations are closed under union, i.e., whenever $\rho, \sigma$ are bisimulations, then so is $\rho \cup \sigma$. This immediately follows by case distinction. $\square$

# Strong Bisimilarity vs. Trace Equivalence

## Theorem

*$P \sim Q$ implies that $P$ and $Q$ are trace equivalent. The reverse does generally not hold.*

## Proof.

The implication from left to right follows from Lemma 4.8.

Consider the other direction:

- Take $P = a.P_1$ with $P_1 = b.\text{nil} + c.\text{nil}$ and $Q = a.b.nil + a.c.\text{nil}$.
- Then: $Tr(P) = \{\epsilon, a, ab, ac\} = Tr(Q)$.
- Thus, $P$ and $Q$ are trace equivalent.
- But: $P \not\sim Q$, as there is no state in the LTS of $Q$ that is bisimilar to $P_1$ (cf. Example 4.6).
- Why? Since no state in $Q$ can perform both $b$ and $c$.

□

# Congruence

**Theorem (CCS congruence property of $\sim$)**

*Strong bisimilarity $\sim$ is a CCS congruence, that is, whenever $P, Q \in Prc$ such that $P \sim Q$,*

$$
\begin{array}{cl}
\alpha.P \sim \alpha.Q & \text{for every } \alpha \in Act \\
P + R \sim Q + R & \text{for every } R \in Prc \\
P \parallel R \sim Q \parallel R & \text{for every } R \in Prc \\
P \setminus L \sim Q \setminus L & \text{for every } L \subseteq A \\
P[f] \sim Q[f] & \text{for every } f : A \to A
\end{array}
$$

# Deadlock Sensitivity of Strong Bisimilarity

**Definition (Deadlock sensitivity; cf. Definition 3.10)**

Relation $\equiv \subseteq Prc \times Prc$ is deadlock sensitive whenever:

$$P \equiv Q \quad \text{implies} \quad (\forall w \in Act^* : P \text{ has a } w\text{-deadlock iff } Q \text{ has a } w\text{-deadlock}) .$$

**Theorem**

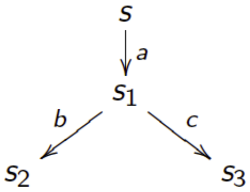
$\sim$ *is deadlock sensitive.*

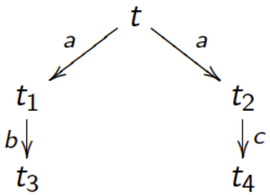
**Proof.**

Let $P \sim Q$.

- We assume that, for some $w \in Act^*$, $P$ has a $w$-deadlock but $Q$ does not.
- Thus, there exists $P' \in Prc$ such that $P \xrightarrow{w} P'$ and $P' \not\longrightarrow$.
- Moreover, for all $Q' \in Prc$ with $Q \xrightarrow{w} Q'$ there exist $\alpha \in Act$ and $Q'' \in Prc$ such that $Q' \xrightarrow{\alpha} Q''$.
- For $P \xrightarrow{w} P'$, Lemma 4.8 (bisimulation on paths) yields $Q'$ with

# How to Show Non-Bisimilarity?




## Alternatives to prove that $s \not\sim t$

- Enumerate all binary relations and show that none of those containing $(s, t)$ is a strong bisimulation.
  - This is expensive, as there are $2^{k^2}$ binary relations on a set $S$ with $|S| = k$.
- Make certain observations which will enable to disqualify many bisimulation candidates in one step.
  - Yields heuristics – how about completeness?
- Use game characterisation of strong bisimilarity.

# The Strong Bisimulation Game

Let $(S, Act, \longrightarrow)$ be an LTS and $s, t \in S$. Question: does $s \sim t$ hold?

We define a game with two players: an "attacker" and a "defender".

- The game is played in rounds, and configurations of the game are pairs of states from $S \times S$.
- In each round, the game is in a current configuration.
- Initially, the configuration $(s, t)$ is chosen as the current one.

## Intuition

The defender wants to show that $s \sim t$ while the attacker aims to prove the opposite.

# Rules of the Bisimulation Game

## Rules

In each round, the current configuration $(s, t)$ is changed as follows:

(1) the attacker chooses one of the two processes in the current configuration, say $t$, and makes an $\xrightarrow{\alpha}$-move for some $\alpha \in \mathit{Act}$ to $t'$, say, and

(2) the defender must respond by making an $\xrightarrow{\alpha}$-move in the other process $s$ of the current configuration under the same action $\alpha$, yielding $s \xrightarrow{\alpha} s'$.

The pair of processes $(s', t')$ becomes the new current configuration.
The game continues with another round.

## Results

(1) If one player cannot move, the other player wins:
- attacker cannot move if $s \not\rightarrow$ and $t \not\rightarrow$
- defender cannot move if no matching transition available

(2) If the game is played *ad infinitum*, the defender wins.

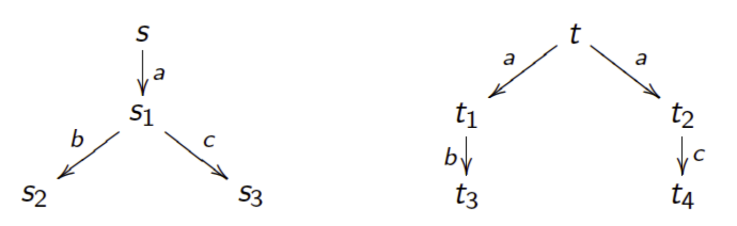
# Examples

## Example 5.1 (Bisimulation games)

(1) Use the CAAL games feature to show $P \sim Q$ where

$$
\begin{aligned}
P &= a.P_1 + a.P_2 & \qquad Q &= a.Q_1 \\
P_1 &= b.P_2 & Q_1 &= b.Q_1 \\
P_2 &= b.P_2 & &
\end{aligned}
$$

(2) Use the CAAL games feature to show that $s \not\sim t$ where




Two winning strategies for attacker in configuration $(s, t)$:

- start with $s \xrightarrow{a} s_1$

# Game Characterisation of Bisimulation

**Theorem 5.2 (Game characterisation of bisimulation) (Stirling 1995, Thomas 1993)**

(1) *$s \sim t$ iff the defender has a universal winning strategy from configuration $(s, t)$.*

(2) *$s \not\sim t$ iff the attacker has a universal winning strategy from configuration $(s, t)$.*

*(By means of a universal winning strategy, a player can always win, regardless of how the other player selects their moves.)*

**Proof.**

by relating winning strategy of defender/attacker to existence/non-existence of strong bisimulation relation □

Thus, a bisimulation game can be used to prove bisimilarity as well as non-bisimilarity.[1] It often provides elegant arguments for $s \not\sim t$.

# Strong Simulation

**Observation:** sometimes, the concept of strong bisimulation is too strong (example: extending a system by new features).

## Definition 5.3 (Strong simulation)

- Relation $\rho \subseteq Prc \times Prc$ is a strong simulation if, whenever $(P, Q) \in \rho$ and $P \xrightarrow{\alpha} P'$, there exists $Q' \in Prc$ such that $Q \xrightarrow{\alpha} Q'$ and $P' \,\rho\, Q'$.
- $Q$ strongly simulates $P$, denoted $P \sqsubseteq Q$, if there exists a strong simulation $\rho$ such that $P \,\rho\, Q$. Relation $\sqsubseteq$ is called strong similarity.
- $P$ and $Q$ are strongly simulation equivalent if $P \sqsubseteq Q$ and $Q \sqsubseteq P$.

**Thus:** If $Q$ strongly simulates $P$, then whatever transition $P$ takes, $Q$ can match it while retaining all of $P$'s options.

**But:** $P$ does not need to be able to match each transition of $Q$!

# Simulation: Example

## Definition (Strong simulation)

- Relation $\rho \subseteq Prc \times Prc$ is a strong simulation if, whenever $(P, Q) \in \rho$ and $P \xrightarrow{\alpha} P'$, there exists $Q' \in Prc$ such that $Q \xrightarrow{\alpha} Q'$ and $P' \,\rho\, Q'$.
- $Q$ strongly simulates $P$, denoted $P \sqsubseteq Q$, if there exists a strong simulation $\rho$ such that $P \,\rho\, Q$. Relation $\sqsubseteq$ is called strong similarity.
- $P$ and $Q$ are strongly simulation equivalent if $P \sqsubseteq Q$ and $Q \sqsubseteq P$.

## Example 5.4

$$\begin{tikzcd}
& P \arrow[dl, "a"'] \arrow[dr, "a"] & \\
P_1 \arrow[d, "b"'] & & P_3 \arrow[d, "c"] \\
P_2 & & P_4
\end{tikzcd}
\qquad
\begin{tikzcd}
& Q \arrow[d, "a"] & \\
& Q_1 \arrow[dl, "b"'] \arrow[dr, "c"] & \\
Q_2 & & Q_3
\end{tikzcd}$$

$Q$ strongly simulates $P$, but not vice versa

This yields that:

$$\begin{array}{lcll} a.b.\text{nil} + a.c.\text{nil} & \sqsubseteq & a.(b.\text{nil} + c.\text{nil}) & \text{and} \\ a.(b.\text{nil} + c.\text{nil}) & \not\sqsubseteq & a.b.\text{nil} + a.c.\text{nil}. & \end{array}$$

(Note that $P \not\sim Q$.)

# Strong Simulation and Bisimilarity

## Lemma 5.5 (Bisimilarity implies simulation equivalence)

*If $P \sim Q$, then $P \sqsubseteq Q$ and $Q \sqsubseteq P$.*

## Proof.

A strong bisimulation $\rho \subseteq Prc \times Prc$ for $P \sim Q$ is a strong simulation for both directions. □

**Caveat:** The converse does not generally hold!

## Example 5.6

$$\begin{tikzcd} & P \arrow[dl, "a"'] \arrow[dr, "a"] & \\ P_1 \arrow[d, "b"] & & P_3 \\ P_2 & & \end{tikzcd} \qquad \begin{tikzcd} Q \arrow[d, "a"] \\ Q_1 \arrow[d, "b"] \\ Q_2 \end{tikzcd}$$

$P \sqsubseteq Q$ and $Q \sqsubseteq P$, but $P \not\sim Q$

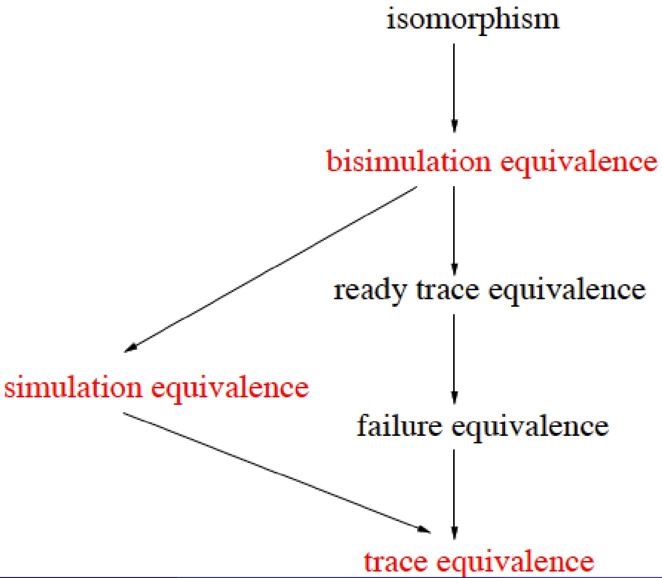
**Reason:** $\sim$ allows the attacker to switch sides at each step!

# Summary: Strong (Bi-)Similarity

## Summary

- Strong bisimulation of processes is based on mutually mimicking each other.
- Strong bisimilarity $\sim$:
  (1) is the largest strong bisimulation
  (2) is an equivalence relation
  (3) is strictly coarser than LTS isomorphism
  (4) is strictly finer than trace equivalence
  (5) is a CCS congruence
  (6) is deadlock sensitive
  (7) can be checked using a two-player game
- Strong similarity $\sqsubseteq$:
  (1) is a one-way strong bisimilarity
  (2) bi-directional version (strong simulation equivalence) is strictly coarser than $\sim$

# Overview of Some Behavioral Equivalences



# Inadequacy of Strong Bisimilarity

## Example 5.7 (Two-place buffers; cf. Example 2.5)

(1) Sequential two-place buffer:

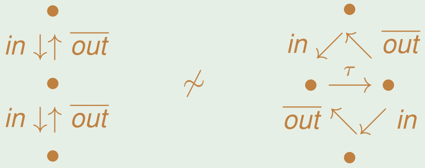
$$\begin{aligned} B_0 &= in.B_1 \\ B_1 &= \overline{out}.B_0 + in.B_2 \\ B_2 &= \overline{out}.B_1 \end{aligned}$$

(2) Parallel two-place buffer:

$$\begin{aligned} B_{\|} &= (B[f] \parallel B[g]) \setminus com \\ B &= in.\overline{out}.B \end{aligned}$$

$(f := [out \mapsto com],$
$g := [in \mapsto com])$

**Observation:**




## Conclusion

- The requirement in $\sim$ to exactly match all actions is often too strong.
- This suggests to weaken this and not insist on exact matching of $\tau$-actions.
- Rationale: $\tau$-actions are special as they are internal and thus unobservable.

# The Rationales for Abstracting from $\tau$-Actions

- $\tau$-actions are internal and thus unobservable.
- This is natural in parallel communication resulting in $\tau$:
  - synchronization in CCS is binary handshaking
  - observation means communication with the process
  - thus the result of any communication is unobservable
- Strong bisimilarity treats $\tau$-actions as any other action.
- Can we retain the nice properties of $\sim$ while "abstracting" from $\tau$-actions?

# Weak Transition Relation

## Definition 5.8 (Weak transition relation)

For $\alpha \in Act$, $\stackrel{\alpha}{\Longrightarrow} \subseteq Prc \times Prc$ is given by

$$\stackrel{\alpha}{\Longrightarrow} := \begin{cases} \left(\stackrel{\tau}{\longrightarrow}\right)^* \circ \stackrel{\alpha}{\longrightarrow} \circ \left(\stackrel{\tau}{\longrightarrow}\right)^* & \text{if } \alpha \neq \tau \\ \left(\stackrel{\tau}{\longrightarrow}\right)^* & \text{if } \alpha = \tau. \end{cases}$$

where $\left(\stackrel{\tau}{\longrightarrow}\right)^*$ denotes the reflexive and transitive closure of relation $\stackrel{\tau}{\longrightarrow}$.

## Informal meaning

- If $\alpha \neq \tau$, then $P \stackrel{\alpha}{\Longrightarrow} P'$ means that from $P$ we can get to $P'$ by doing zero or more $\tau$ actions, followed by the action $\alpha$, followed by zero or more $\tau$ actions.
- If $\alpha = \tau$, then $P \stackrel{\alpha}{\Longrightarrow} P'$ means that from $P$ we can reach $P'$ by doing zero or more $\tau$ actions.

# Weak Bisimulation

## Definition 5.9 (Weak bisimulation) (Milner 1989)

A binary relation $\rho \subseteq Prc \times Prc$ is a weak bisimulation whenever for every $(P, Q) \in \rho$ and $\alpha \in Act$ (including $\alpha = \tau$):

(1) if $P \xrightarrow{\alpha} P'$, then there exists $Q' \in Prc$ such that $Q \stackrel{\alpha}{\Longrightarrow} Q'$ and $P' \, \rho \, Q'$, and

(2) if $Q \xrightarrow{\alpha} Q'$, then there exists $P' \in Prc$ such that $P \stackrel{\alpha}{\Longrightarrow} P'$ and $P' \, \rho \, Q'$.

## Definition 5.10 (Weak bisimilarity)

Processes $P$ and $Q$ are weakly bisimilar, denoted $P \approx Q$, iff there is a weak bisimulation $\rho$ with $P \, \rho \, Q$.

$$\approx \; = \bigcup \{\rho \subseteq Prc \times Prc \mid \rho \text{ is a weak bisimulation}\}.$$

Relation $\approx$ is called weak bisimilarity or observational equivalence.

# Explanation

**Definition (Weak bisimulation)** (Milner 1989)

A binary relation $\rho \subseteq Prc \times Prc$ is a weak bisimulation whenever for every $(P, Q) \in \rho$ and $\alpha \in Act$ (including $\alpha = \tau$):

(1) if $P \xrightarrow{\alpha} P'$, then there exists $Q' \in Prc$ such that $Q \overset{\alpha}{\Longrightarrow} Q'$ and $P' \,\rho\, Q'$, and

(2) if $Q \xrightarrow{\alpha} Q'$, then there exists $P' \in Prc$ such that $P \overset{\alpha}{\Longrightarrow} P'$ and $P' \,\rho\, Q'$.

**Remark**

Each clause in the definition of weak bisimulation subsumes two cases:

- $P \xrightarrow{\alpha} P'$ where $\alpha \neq \tau$:
  There exists $Q' \in Prc$ such that $Q \,(\xrightarrow{\tau})^* \xrightarrow{\alpha} (\xrightarrow{\tau})^*\, Q'$ and $P' \,\rho\, Q'$.
- $P \xrightarrow{\tau} P'$:
  There exists $Q' \in Prc$ such that $Q \,(\xrightarrow{\tau})^*\, Q'$ and $P' \,\rho\, Q'$ (where $Q' = Q$ is admissible).

## Example 5.11

(1) Let $P = \tau.Q$ with $Q = a.\text{nil}$.

- obviously $P \not\sim Q$; claim: $P \approx Q$
- proof: $\rho = \{(P, Q), (Q, Q), (\text{nil}, \text{nil})\}$ is a weak bisimulation with $P \, \rho \, Q$

(2) More general: for every $P \in Prc$, $P \approx \tau.P$.

Proof: $\rho = \{(P, \tau.P)\} \cup id_{Prc}$ is a weak bisimulation:

- every transition $P \xrightarrow{\alpha} P'$ can be simulated by $\tau.P \xrightarrow{\tau} P \xrightarrow{\alpha} P'$ (i.e., $\tau.P \xRightarrow{\alpha} P'$)
  with $P' \, \rho \, P'$ (since $id_{Prc} \subseteq \rho$)
- the only transition of $\tau.P$ is $\tau.P \xrightarrow{\tau} P$; it is simulated by $P \xrightarrow{\tau}^{0} P$ with $P \, \rho \, P$ (since $id_{Prc} \subseteq \rho$)

(3) Sequential and parallel two-place buffer are weakly bisimilar (check with CAAL):

$$\begin{array}{c} P_1 \\ in \downarrow\uparrow \overline{out} \\ P_2 \\ in \downarrow\uparrow \overline{out} \end{array} \qquad \begin{array}{ccc} & Q_1 & \\ in \swarrow & & \nwarrow \overline{out} \\ Q_2 & \xrightarrow{\tau} & Q_3 \\ \overline{out} \nwarrow & & \swarrow in \end{array} \qquad \rho = \{(P_1, Q_1), (P_2, Q_2), (P_2, Q_3),$$